



DELIVERABLE

Project Acronym: EAGLE

Grant Agreement number: 325122

Project Title: Europeana network of Ancient Greek and Latin

Epigraphy

The EAGLE Flagship Storytelling Application D5.4.2

version: N° 1.0

Revision: final

Authors:

Wolfgang Schmidle (DAI) Francesco Mambrini (DAI) Philipp Franck (DAI)

Contributors:

Vittore Casarosa (CNR-Isti) Pietro Liuzzo (University of Heidelberg) Claudio Prandoni (Promoter srl)

Reviewers:

László Borhy (ELTE) Rada Varga (UBB) Anja Ragolič (ZRC SAZU)

| Project co-funded by the European Commission within the ICT Policy Support Programme | | | |
|--------------------------------------------------------------------------------------|------------------------------------------------------------------------------|---|--|
| Dissemination Level | | | |
| Р | Public | x | |
| С | Confidential, only for members of the consortium and the Commission Services | | |



Revision History

| Revision | Date | Author | Organisation | Description |
|----------|------------|--------------------|--------------|----------------------------------------------------------------------------------------|
| 0.1 | 05/10/2015 | Wolfgang Schmidle | DAI | First draft, based on D5.4.1, reworked for version 2.0 of the Storytelling Application |
| 0.2 | 06/10/2015 | Francesco Mambrini | DAI | Updated the introduction |
| 0.3 | 08/10/2015 | Philipp Franck | DAI | Proofread the technical descriptions |
| 1.0 | 12/10/2015 | Wolfgang Schmidle | DAI | Updated the figures, integrated the peer reviewers' comments |

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.



TABLE OF CONTENTS

| LIST OF ABBREVIATIONS | 5 |
|------------------------------------------------------------------------------------|----|
| EXECUTIVE SUMMARY | 6 |
| 1. INTRODUCTION | |
| | |
| 1.1. DOCUMENT PURPOSE | |
| 1.2. DESIGN AND RATIONALE | |
| 1.2.1. Purpose of the FSA | |
| 1.2.2. Intended audience 1.2.3. Users and stories: read, write and publish | |
| 1.2.4. Development cycle and functional requirements | |
| | |
| 2. USER MANUAL | |
| 2.1. THE FSA ACCESS PAGE | |
| 2.1.1. The main column | |
| 2.1.2. The sidebar | |
| 2.1.3. Searching and filtering the collection of stories | |
| 2.2. THE STORY PAGES | |
| 2.3. THE STORY EDITOR. | |
| 2.3.1. The list view | |
| 2.3.2. The edit view | |
| 2.3.3. The WYSIWYG editor | |
| 2.3.4. The sidebar | |
| 2.3.5. Publishing a story | |
| 2.3.7. Searching inscriptions and other data sources | |
| 3. DEVELOPER'S GUIDE | |
| | |
| 3.1. Integration with the eagle portal | |
| 3.1.1. Two plugins | |
| 3.1.2. The Eagle Storytelling Application plugin | |
| 3.1.3. The Eagle Storytelling Application Bridge plugin | |
| 3.1.4. The "story" post type | |
| 3.2. THE FSA ACCESS PAGE | |
| 3.2.1. Design rationale | |
| 3.2.2. The sidebar | |
| 3.2.3. Searching and filtering stories | |
| 3.3. THE STORY PAGES | |
| 3.4.1. Searching EAGLE inscriptions and other data sources | |
| 3.4.2. Shortcodes | 25 |
| 3.4.3. The separate story editor | |
| 4. CONCLUSIONS AND FUTURE PLANS | |
| | |
| 4.1. DESIGN GOALS IN VERSION 2.0 AND BEYOND | |
| 4.1.1. Consistency | |
| 4.1.2. Encapsulation | |
| 4.1.3. Ease of use 4.1.4. Findability of stories | 28 |
| ד. ו.ד. ווועשווונץ טן אנטווכא | |



| 4.1.5 | 5. Sustainability | 30 |
|---------|-------------------|----|
| 4.2. VI | /ERSION 3.0 | 30 |
| 5. RFFF | FRENCES | 32 |



LIST OF ABBREVIATIONS

FMA EAGLE Flagship Mobile Application

FSA EAGLE Flagship Storytelling Application

For the deliverables D5.1, D5.2, D5.4, D5.5, D5.6 and D6.1 see the References



EXECUTIVE SUMMARY

The Deliverable 5.4.2 introduces the second release of the EAGLE Flagship Storytelling Application (FSA) to the partners and the general public.

The Introduction (Section 1) summarizes some aspects of the design and development cycle of this and other EAGLE applications that are useful to contextualize the work on the FSA.

Section 2 illustrates the three main interfaces of the FSA: the first interface allows users to access the collection of the published stories, the second interface allows users to read each story in full; the third interface provides registered users with a unique storytelling writing environment, where they can compose epigraphy-based narratives and enrich them with context from the EAGLE portal and other online repositories. Section 2 serves also as a user reference manual.

The developer's guide (section 3) details the specifics for the implementation of the FSA services and its interfaces. The FSA is implemented as a pair of plugins in the EAGLE portal to ensure a tight integration of services.

Conclusions and future plans are discussed in Section 4.



1. INTRODUCTION

1.1. DOCUMENT PURPOSE

This document describes version 2.0 of the EAGLE Flagship Storytelling Application (**FSA**). It provides both a **user manual** (section 2) and a **developer's guide** (section 3).

In the following sections of this introduction we summarize some of the preliminary design aspects (target users, functional requirements, development and evaluation cycle) that are discussed elsewhere in detail (see especially D5.1, D5.2, D5.3.1, D5.5). Such details are presented here because they provide an indispensable context for understanding the place of the FSA within the development process of the EAGLE user-dedicated services, as well as important background information for evaluating this first release.

1.2. DESIGN AND RATIONALE

1.2.1. Purpose of the FSA

The FSA is a web-based tool designed to allow users to create multimedia narratives on epigraphic content and share them on the web. It is one of the components of the EAGLE portal (see deliverable D5.2). As such, its interfaces are designed to be integrated within the general architecture of the EAGLE portal, and they are visually coherent with the graphic identity of the EAGLE portal (see deliverable D6.1).

With the help of a friendly interface, users will be able to write texts and include content in different format into their narratives. In particular, the FSA is designed so as to help authors with special interest in epigraphy and in ancient history or archaeology to get easy access to the main resources for multimedia content on the Ancient World. With the FSA we have created a tool that enables users to include images and videos from popular hosts (such as YouTube or Europeana), excerpts from Ancient texts, maps locating ancient places and, especially, the vast collection of inscriptions from the EAGLE network.

The FSA is thus a useful resource to spread the knowledge about monuments and inscriptions in the collections of the EAGLE network, or to suggest unexplored paths through themes and objects. All the users of the EAGLE services, the members of the EAGLE Best Practices Network and the whole community of persons interested in cultural heritage and history, whether specialists or amateurs, are encouraged to unleash their creativity and experiment with the potential of this tool.

1.2.2. Intended audience

The FSA is designed to assist the creativity of our public. Everyone that has a story to tell about inscriptions in the EAGLE collections is a potential user of the application; everyone is encouraged to experiment with this new form of scientific communication. In particular, however, we have in mind two areas where professionals can profit from the tool.

In education, the FSA may especially suit different professional figures and contexts. Stories featuring textual as well as visual content can be used as a support to teaching activities of lecturers and professors. For students, epigraphy-based stories may be an innovative format for essays, assignments and presentations. Scholars and researchers, on the other hand, can use the FSA as a "blogging" platform to disseminate preliminary results or, more generally, to inform the public about ongoing research activities.

Cultural heritage professionals are also strategic figures to engage. The FSA is potentially an excellent medium that curators of museums and exhibitions can use to promote permanent collections and special events. Narratives based on single objects can leverage the multimedia capabilities of the FSA to provide information on the geographical and the historical context of each monument. Maps, photographs and



passages from ancient texts (either in the original and/or in translation), and videos can be interwoven with textual explanations, so that the public can get an easier and visually stimulating access to information about the objects on display.

On the other hand, stories built as "galleries" of different monuments or inscriptions can be an attractive way to promote, advertise or introduce the public to new initiatives and exhibitions. Narratives that are easily accessible on the web and include written descriptions as well as visual content are the ideal medium to suggest possible thematic paths through collections to the attention of the audience.

The deliberate decision to cater for the general public as well as the scientific community will naturally lead to stories with different target audiences. In chapter 4 we will discuss some strategies to direct different groups of readers to the stories they will be interested in the most.

1.2.3. Users and stories: read, write and publish

Examples of the possible stories that can be shared thanks to the FSA include notes centered on one specific monument, where the multimedia content can be effectively used to contextualize the object. Or stories can be used to guide the readers on new thematic journeys, by collecting and discussing all the inscriptions from a certain area, or dated to a specific era, or important for specific topics, like e.g. slavery, and so on.

On a dedicated page on the EAGLE portal, the public has access to the stories written and published using the FSA. In addition to reading what others have shared, users can easily start to write and publish their own content. Every user can register a personal account on the EAGLE portal that grants access to the unique features of all the EAGLE User Dedicated Services (including also dedicated features of the EAGLE Portal and the Flagship Mobile Application), as discussed in the Deliverables D5.1, D5.2 and D5.3.

Users with registered EAGLE accounts can immediately create their own stories. The interface is designed so as to make the process of writing, searching and including multimedia content as fluent and intuitive as possible. With the press of a button authors can embed the retrieved objects (text excerpts, videos, images and inscriptions) to the intended location within the story.

1.2.4. Development cycle and functional requirements

The development cycle of the user-dedicated services is outlined in D5.1. For the two flagship applications (the mobile app, on which see D5.3, and the FSA) partners and developers of the EAGLE consortium have scheduled two main releases:

- First release (version 1.0): to be completed before month 21 of the project
- Second release (version 2.0): to be completed before month 30 of the project

This two-step process is also reflected on the priority scale assigned to the functional requirements, as discussed in D5.1: 28, and D5.5: 7. Between the two releases, a functional validation has taken place, see D5.6.1. This validation, together with the feedback of early users of the FSA, has greatly helped the developers to address some bugs and missing features in version 1.0.

Table 1 reports the functional requirements of the FSA as defined in the D5.1 as well as their implementation status in version 2.0.

In the process of creating the FSA, some original requirements were adapted to reflect the fact that the FSA was implemented as a part of the EAGLE WordPress installation, and to allow a better user experience. In particular, the search is implemented in a modal window due to space limitations (STE11), and drag and drop is replaced by a button that is intuitive to use and makes more sense than drag and drop in the modal window (STE12). Some of these deviations from the original plan have been discussed



already in D5.6.1.

| ID | Sec. | User | Requirement | Priority | Status |
|-------|-------|------------|------------------------------------------------------------------------------------------|----------|---------------------|
| STE01 | 4.4.1 | Generic | Perform full text search on stories | High | implemented |
| STE02 | 4.4.1 | Generic | Browse stories with a keyword word-cloud | Medium | implemented |
| STE03 | 4.4.1 | Generic | List of recently published stories | High | implemented |
| STE04 | 4.4.1 | Generic | Share stories on social networks | Low | implemented |
| STE05 | 4.4.2 | Registered | Create new stories | High | implemented |
| STE06 | 4.4.2 | Registered | Create paragraph in WYSIWYG editor | High | implemented |
| STE07 | 4.4.2 | Registered | Use title and description paragraph | High | implemented |
| STE08 | 4.4.2 | Registered | Retrieve and add content from the EAGLE database and from selected external repositories | High | implemented |
| STE09 | 4.4.2 | Registered | Retrieve and add content from the Perseus Project | Medium | planned for v3.0 |
| STE10 | 4.4.2 | Registered | Format of the search results | High | implemented |
| STE11 | 4.4.2 | Registered | Single window for the story editor interface | High | implemented |
| STE12 | 4.4.2 | Registered | Drag and drop content | High | implemented |
| STE13 | 4.4.2 | Registered | Save drafts | High | implemented |
| STE14 | 4.4.2 | Registered | Publish a story | High | implemented |
| STE15 | 4.4.2 | Registered | Edit, unpublish, delete published stories | High | implemented |
| STE16 | 4.4.2 | Registered | Access and include saved items | Medium | planned for v3.0 |
| STE17 | 4.4.2 | Registered | Generic EpiDoc XML reader | Medium | implemented |
| STE18 | 4.4.2 | Registered | Add tags to a story | Medium | implemented |

Table 1: EAGLE Flagship Storytelling Application requirements and implementation status

Two requirements are still missing: Accessing saved items (STE16) will be implemented soon, and Perseus as a data source (STE09) will be implemented as soon as the Perseus is stable enough. Both requirements will be discussed in more detail in section 4.2.

In its current state, the FSA is already perfectly adquate to fulfil its intended goals. Nonetheless, we plan to deliver a final version 3.0 of the FSA by the end of the project that was not foreseen in the original plan. Changes from version 2.0 will be mainly due to the expected feedback from the more widespread use of the FSA after its official launch on 24 September 2015 at the sixth EAGLE International Event. For example, we intend to implement more datasources (see STE08) based on the wishes of the users.



2. USER MANUAL

The EAGLE Flagship Storytelling Application (FSA) includes three interfaces:

- 1. The **FSA** access page, where all users (registered and unregistered) can browse and search the collection of published stories
- 2. The **story pages** for displaying single stories
- 3. The **story editor**, where authors (after logging in) can start composing their own story and edit existing stories

All three interfaces are part of the EAGLE website. The FSA is accessible via the RESOURCES tab in the main menu of the EAGLE website, which leads to the FSA access page. The FSA access page in turn contains links that lead to the story pages and to the story editor. The story pages and the story editor can also be accessed directly if the user knows the direct link.

In the following sections the functionalities of the three interfaces are described in detail. The description will refer to the requirements STE01 to STE18 that are listed in table 1 and laid out in more detail in section 4.4 of the Deliverable D5.1.

2.1. THE FSA ACCESS PAGE

The FSA access page (http://www.eagle-network.eu/stories/) with the title "STORIES" is the first interface of the FSA that the user will see. It has the same look as most other pages in the EAGLE website. In particular, it uses the standard header and footer for EAGLE pages, there is a main column for the content, and when the browser window is wide enough there is an additional sidebar on the right for so-called widgets that can perform functions like searching stories etc.

The display of the story list in the main column resembles the display of News items. The selection of widgets is specific to the FSA, however. The sidebar contains a general "LOGIN" widget, but it includes also some story-centric widgets.

2.1.1. The main column

The main column shows a list of all published stories, starting with the most recent ones. When the user has performed a story search (see section 2.1.3), the main column displays the search results instead.

All stories are listed in the same fashion as the News items, i.e. in a paginated list where each story is represented by its title, a thumbnail image (if the story has one), author name and keywords, and a short excerpt consisting of the first few lines of text with all formatting removed or an explicit abstract. the author and the keywords are clickable links, which amount to the same as choosing the author or keyword in the search filters in the sidebar (see section 2.1.2). Each story has a "Read Story" link that leads to the story page for this story (see section 2.2). In addition, if any keywords are associated with a story they are displayed below the title.

The list of stories implements requirement STE03, which states that the list of most recently published stories should include (1) a limited number of stories, e.g. between 5 and 10, and for each story the list should display (2) the title, (3) the author's user name and (4) the text of the description paragraph.



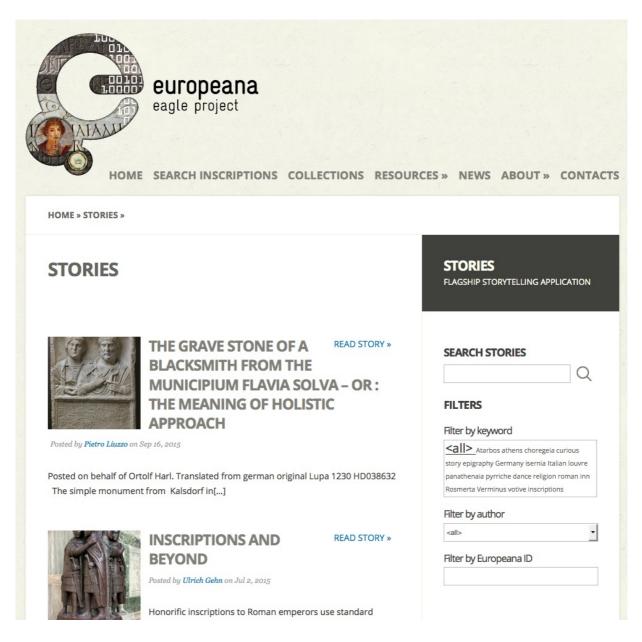


Figure 1: The FSA access page

2.1.2. The sidebar

The sidebar starts with a box "STORIES", which takes the user from any FSA page back to the FSA access page. The next widgets, "SEARCH STORIES" and "FILTERS", are described in the following section. Below the search widgets are the "LATEST STORIES" widget and the "LOGIN" widget.

The "LATEST STORIES" widget is a more concise version of the story list in the main column. Only the titles of the ten most recent stories are displayed and are links to the story pages of these stories (see section 2.2). This widget allows for more consistency with the overall design of the EAGLE website, especially the News page, and it remains in place even when the main column displays a search result rather than the full list of stories.



In the "LOGIN" widget the user can log in using their normal EAGLE user account. In order to save some screen real estate, this is actually a link to dedicated login page. After login, the user is automatically redirected back to the FSA access page. They will now see a link "Create new story", which leads to the edit view of the story editor, and a link "Edit my stories", which leads to the list view of the story editor where the user can see their published and unpublished stories (see section 2.3).

LOGGED IN AS WOLFGANG

Edit this story Create new story Edit existing stories

Logout

Figure 2: "LOGIN" widget when logged in

2.1.3. Searching and filtering the collection of stories

The FSA-specific widgets "SEARCH STORIES" and "FILTERS" help the user navigate through the collection of published stories.

The widget "SEARCH STORIES" is dedicated to searching the full text of all published stories. If the user types in one or more words in the search field, the search result (i.e. all stories that fit this request) will be displayed in the main column. For example, a search for the German word "Grabinschrift" will find all stories that include this word. However, it will not find stories that only include the English counterpart "funerary".

The second widget is called "FILTERS". Regardless of the actual content of the main column (all stories or a search result), the user can restrict the displayed list to stories that satisfy one or more additional filter criteria. The widget is simply a wrapper for a group of filter widgets, namely "Filter by keyword", "Filter by author", "Filter by Europeana ID" and "Language", and doesn't do anything in itself. In the following we will refer to the "Filter by …" widgets either as filter widgets or as sub-widgets of "FILTERS".

With the sub-widget "Filter by keyword", the user can choose to see only stories that contain a given keyword by clicking on it in the keyword cloud. For example, the author of a story may have assigned the keyword "funerary" to it. Thus, the story can be found even if the full text of the story is written in German and does not contain the word "funerary".

In the sub-widget "Filter by author" the user can choose an author, or more precisely the user name of the author, from a dropdown menu of the most active authors and then will see only stories by this author. If the user is logged in, they will have an additional option "show only my stories". This will have the same effect as if the user had chosen their own user name in dropdown menu. The main difference to the "edit my stories" link in the "LOGIN" widget is that the user only sees their published stories, but not their story drafts.

In the sub-widget "Filter by Europeana ID" the user can filter the search result for stories that contain inscriptions with a given Europeana ID. This includes all items from the EAGLE database.

Together the "SEARCH STORIES" and "FILTERS" widgets implement the requirement STE01, which states that users should be able to (1) perform full-text searches of all published stories and to search for (2) keywords set by the author, (3) the user name of the author and (4) the Trismegistos IDs of the EAGLE inscriptions included in the story. (For a discussion why Trismegistos was replaced by Europeana



see section 4.1.4.) In addition, the word cloud implements the requirement STE02.

| SEARCH STORIES | LATEST STORIES | | |
|------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|--|
| FILTERS | The grave stone of a blacksmith from the Municipium Flavia Solva – or: The Meaning of holistic approach | | |
| TIETERS | Inscriptions and beyond | | |
| Filter by keyword | How many lives could an inscription live? | | |
| <u><all></all></u> Atarbos athens choregeia curious | A Silver Statue and a Golden Mouth | | |
| story epigraphy Germany isernia Italian louvre panathenaia pyrriche dance religion roman inn Rosmerta Verminus votive inscriptions | 1 fragmentary epigraph, 1 anonymous relief fragment, and 2 gods | | |
| Filter by author | Herculanus, Auggustorum nostrorum verna | | |
| <all> ▼</all> | Where are the Dacians? | | |
| Filter by Europeana ID | The wax tablets of Alburnus Maior | | |
| пист ву сегореснить | The Building inscriptions of Eleusis and the importance of the epigraphic evidence for reconstruction of architectural remains | | |
| | Tile messages from Valencia | | |

Figure 3: Fulltext search with filters, and Latest stories

2.2. THE STORY PAGES

Each story has its own story page, just as each entry in a blog can appear in the list of all blog posts as well as on its own page. A story page displays a single story. Each story page has a different title, namely the title of the story it displays.

In the main column the story is shown in its entirety with all its text formatting and all images. Below the story title are the author, the post date and keywords associated with this story. Author and keywords are clickable links. Next to the autor and keywords there is a "SHARE" link for sharing the story on Facebook, Twitter, etc. This implements requirement STE04, which states that users should be able to share their stories on social networks.

The story itself consists of an excerpt (if there is one) in italics but without any other formatting, and then the main story text. There is no need for a "Read more" link. However, as on the FSA access page there will be a link "Edit this story" if the author of the story is logged in.

There are three ways to get back from a story page to the FSA access page. First, one can of course use the back button of the browser. As can be expected, this will lead to the last search results if there are any. Second, the breadcrumb "HOME " STORIES " (Story title)" is a clickable link. And third, there is a clickable box "STORIES, Flagship Story Application" at the top of the sidebar.

The sidebar is the same as on the FSA access page. "LOGIN" and "LATEST STORIES" behave exactly the same as on the FSA access page, with the obvious difference that the user is now redirected back to the story page after logging in. The "SEARCH STORIES" and "FILTERS" of course lead to the search result on the FSA access page.



FUNERARY INSCRIPTIONS IN THE CURRENT EAGLE COLLECTION

Posted by admin on Jan 29, 2015

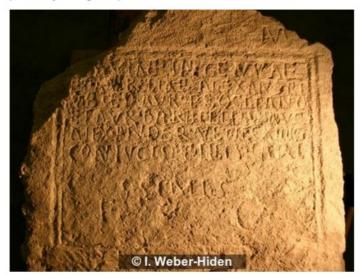
SHARE »

To get a synoptic view of all the inscriptions with funerary destination that are currently available in the EAGLE collection we will use the Type of Inscription field of the Advanced Search. This field actually triggers a menu where the preferred labels in the EAGLE controlled vocabulary are reported; the user has only to pick the desired value in order to perform the query.

In fact, this is not an easy task. The preferred labels are not in a single language, and the value that one looks for (e.g. the equivalent of type "funerary") can be expressed in any of the languages in use of the content providers of EAGLE. In our specific case, for example, the language is Latin: if we pick the type "Sepulcralis" from the Advanced Search options we can extract all the inscription that are described as e.g. "funerary" or "Grabinschriften".

At the moment, such a query returns 56220 results. It might be interesting to compare this datum with the numbers of all other types attested (or at least the major ones, like e.g. "votive", "honorary inscriptions" etc.). We could display a pie chart that can easily be inserted into this story!

Among the results that are returned, the inscription dedicated by the veteran Marcus Aurelius Alexander to his wife Cavia Ingenua and sons stands out immediately as a particularly striking example.



The text, as edited in the Epigraphic Database of Heidelberg, reads:

[D(is)] M(anibus) / [C]aviae Ingenuae / et Aureliae Alexandri/ae et Aur(elio) Procliano / et Aur(eliae) Barbillae M(arcus) Aur(elius) / Alexander vet(eranus) leg(ionis) XIIII / coniugi et filis caris/simis

Figure 4: Display of (the first half of) a single story



2.3. THE STORY EDITOR

The story editor is accessible only to users that are logged in. It has two views, namely the **list view** and the **edit view**. In the current version 2.0 of the FSA, both views are based on the corresponding views of the blog post editing functionality in WordPress¹. However, the views are specifically adapted for editing stories.

2.3.1. The list view

In the list view, the user sees a list of all stories they have created. This includes not only published stories, but also unpublished story drafts. The stories are listed in a more concise way than on the FSA access page. While the FSA access page is made for browsing stories and finding stories that are interesting, the list view should make it easy to locate any particular story the user wants to edit. For this reason the story titles are listed without any keywords and without any excerpt of the main text.

Each story has a link "Edit this story", and there is one link "Create new story". These links lead to the edit view. In the case of "Create new story", the user starts with an empty story. For each story there is also a link "Delete this story".

In the list view there are no widgets in the sidebar.

2.3.2. The edit view

In the edit view, the user can create a new story or edit existing ones, regardless of their publication status. The publication status can be changed in the "Publish" widget in the sidebar (see section 2.3.5).

Together, the two views implement requirement STE05, which states that (1) logged in users can create new stories, and that (2) the stories will be visible to other users only after they are published.

The following sections describe aspects of the edit view in more detail: The WYSIWYG editor, the widgets in the sidebar and the modal window in which one can search different data sources.

2.3.3. The WYSIWYG editor

The centrepiece of the edit view is the WYSIWYG ("what you see is what you get") editor. WordPress already contains a WYSIWYG editor that can be leveraged for the creation of stories.

The WYSIWYG editor provides all the standard features that one would expect. The user must specify the title of the story in a slot at the obvious position above the main text entry field. In the main text entry field the user can structure the main text of the story into paragraphs and add section headings at the appropriate places. The editor supports text formatting features such as bold, italic and underlined.

A widget "Excerpt" below the main text entry field allows the user to add a description paragraph about the story. If an excerpt is added, the abbreviated view on the FSA access page will automatically display exactly this excerpt instead of cutting off the text at an arbitrary point. Strictly speaking, the "Excerpt" widget is not part of the WYSIWYG editor. As in the title, the formatting features of the WYSIWYG editor can't be used in the excerpt, and the description paragraph will automatically be formatted in italics.

The built-in WordPress WYSIWYG editor provides all the requirements in STE06, which states that there should be (1) a WYSIWYG editor that can create (2) text paragraphs and (3) section headers and can (4) mark text as bold, italic and underlined. It also provides the requirements in STE07, which adds that stories should have (5) a title and (6) a description paragraph.

¹ https://wordpress.org/



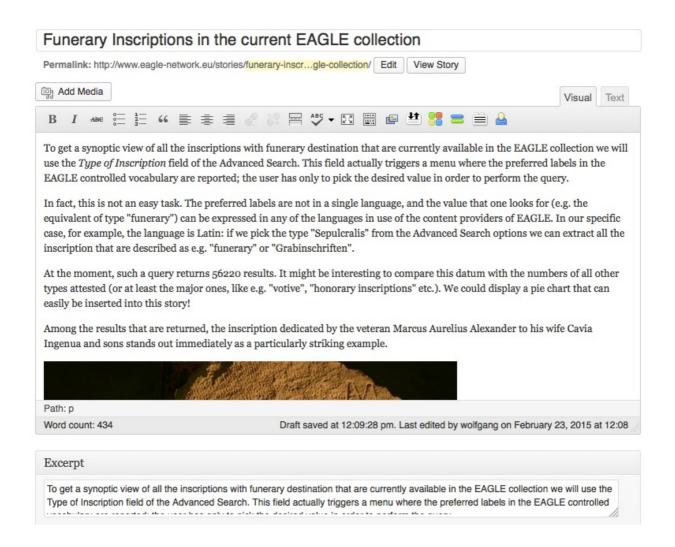


Figure 5: The WYSIWYG editor

2.3.4. The sidebar

The sidebar contains a widget "Keywords" where the story author can add one or more keywords to the story to make it more easily accessible to user searching for interesting stories. This widget is the counterpart of the "Filter by keyword" widget described in section 2.1.3. The "Keyword" widget implements STE18, which states that users should be able to enter free-text tags to classify the content of their stories.

The other two "Filter by" widgets do not need a counterpart in the story editor. For the "Filter by author" widget this is obvious since the author is a fixed piece of information attached to each story. The "Filter by Europeana ID" widget does not need a counterpart because the filter finds all stories that contain items with a given ID (see section 2.3.7). Likewise, the "LOGIN" widget is not needed in the story editor interface, but WordPress provides a possibility to log out.



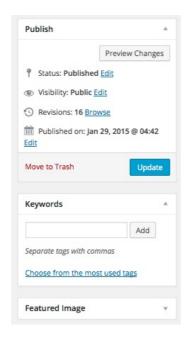


Figure 6: The edit view sidebar

2.3.5. Publishing a story

In the sidebar users will also find the "Publish" widget. Here they can save draft versions, publish the story, change the publication status back to draft, and even delete the story. Publishing a story adds it to the list of published stories in the FSA access page (see section 2.1).

When the story has been published or saved as draft, the user can always resume working on the story simply by going to the story editor page for this story again. If they work on an already published story, the "Publish" button changes to "Update". If they do not update, the story editor will save the published version as well as the draft.

This implements the requirements STE13, STE14 and STE15. Requirement STE13 states that users should be able to (1) save drafts of their stories and to (2) resume their work and edit the drafts at any time. STE14 adds that (3) users should be able to publish their stories. STE15 further adds that users should be able to (4) edit, (5) unpublish, or (6) delete the stories they have published.

2.3.6. The EAGLE media library

In the story editor there is a standard WordPress "Add Media" modal window. When the user clicks on the "Add Media" button, the modal window will appear. In this modal window the user has access to the standard EAGLE media library.

The EAGLE media library has two tabs, "Upload Files" and "Media Library". If the story author wants to include an image in their story, the image needs to be uploaded to the EAGLE media library first (tab "Upload Files"). In the second step this or any other image from the media library can be inserted into the story by selecting the image and clicking the button "Insert into post". The modal window will then disappear, and the image will be inserted at the current position of the cursor.



2.3.7. Searching inscriptions and other data sources

The search of external data sources is implemented as part of the same "Add Media" modal window. The FSA-specific search of external datasources can be accessed by clicking on "EAGLE Storytelling Application" in the left column. The story author can then choose from a variety of data sources. The procedure is always similar: The user interface is a simple search field and possibly some additional filters. The items in the result list are displayed as rectangles. The items are shown in the same way as they would appear in the story if the user decided to insert them. Each item is shown in a default "short view", which can be changed to its "long view" by clicking at the bottom of the rectangle. Another click leads back to the short view.

An item can be selected by clicking on its rectangle. It will then be displayed on the right hand side and can be inserted into the story at the current cursor position by clicking on the button "insert into post". Additional information such as a desired width or height can be entered. This information can later be changed by editing the shortcode (see section 3.4.2) of the item in the text.

In the rectangles, some additional options are visible on mouseover. The options depend on the situation: In the search interface there is a link to the display of the item in the original data source. In the story editor, i.e. when the item was inserted into the story text, one can either delete this item from the story or set the image (if there is one) as the thumbnail of the story, which is displayed on the list of stories on the FSA access page. On a single story page, there is again a link to the item in the original data source, and a link "Find stories with the same item".

The FSA aims for implementing the simple search capabilities of the original data sources, but not their advanced search capabilities. Whenever advanced search capabilities of the original data source are needed, one can find an item in the original data source and then copy the URL from the data source into the search field instead of searching for it within the FSA. The system will then display this item, and it can be inserted into the story text. The difference to pasting the URL directly into the story text is that the item will be inserted as a shortcode rather than the simple URL and displayed as a rectangle rather than a link.

A number of data sources has already been implemented:

EAGLE: There are no additional filters. If the EAGLE item is an inscription in the EpiDoc format, it will be rendered correctly. (This implements STE17, Generic EpiDoc reader). Due to some limitations on the EAGLE side it is not possible to paste EAGLE item URLs into the system.

Europeana: The additional filters are the media types in Europeana and a checkbox to restrict the search to EAGLE items in Europeana.

iDAI.gazetteer: There are no additional filters, just as in the simple search of the original data source. The iDAI.gazetteer distinguishes between the URL representing a *place* and the URL of the electronic *representation* of the place in the system. The FSA accepts both types of URLs.

Wikipedia: There are no additional filters. A search is carried out in the English Wikipedia. However, the system accepts Wikipedia URLs in all languages.

YouTube: Unlike the other datas sources, YouTube videos can be embedded directly into the text via their URLs. There is no need to go through the "Add media" modal window.

More data sources will be implemented in the course of the next months.



The search functionality in the story editor implements the following requirements:

- STE08: The editor interface should allow users to search for objects from the EAGLE collection and external repositories, in particular from Europeana, Wikipedia, YouTube, Flickr and Pelagios. (Flickr and Pelagios will be implemented soon.)
- STE10: Search results should be formatted as: (1) a thumbnail image, (2) a title, positioned on the top-right side of the image, (3) basic information about the object below the title. The requirements were formulated before the EAGLE search had been implemented, and after some further discussions the decision was made that the EAGLE search results should be presented in exactly same way as on the dedicated EAGLE search page, i.e. the same information about the objects should be displayed. This supersedes the original requirement at least in the case of the EAGLE inscription search.
- STE11 (users should be able to write a story, search for objects and see the result list on a single window) and STE12 (the editor interface should have a drag&drop functionality for objects and text paragraphs in the search results): The implementation of the search in external data sources is a change from requirement STE11, which states that users should be able to write a story, search for objects and see the result list on a single window, as seen in services such as Storify. However, there is a good reason for this change. In the standard page layout of the EAGLE portal there is simply not enough screen real estate to provide search functionality within the story editor page. Thus, the modal window is a compromise that implements not the letter but the spirit of the requirement. Besides, the same solution is already used for other data sources in "Add media" modal window. The change from drag&drop to the standard "Insert into post" button is a consequence of using the modal window.

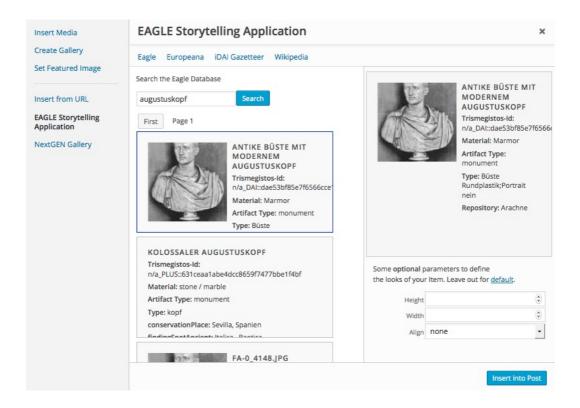


Figure 7: The "Add media" modal window



3. DEVELOPER'S GUIDE

In a nutshell, the Flagship Storytelling Application (FSA) is a pair of plugins in the EAGLE WordPress installation. Stories are blog posts with a specific post type called "story". The FSA access page is a WordPress page with its own template file. The story pages are comparable to single blog post pages and also have their own template file. The story editor with its list view and edit view is a modified clone of the WordPress blog post editor. All of this is described in detail below.

The version numbers 1.0 and 2.0 reflect the two deadlines for the FSA in the EAGLE project. Internally the FSA is being continually worked on, and new features in the test environment are regularly deployed in the productive system after some testing. Since this Deliverable can only capture a certain moment in time, some features in the FSA may differ from the description in the Deliverable.

3.1. INTEGRATION WITH THE EAGLE PORTAL

The Flagship Storytelling Application (FSA) is implemented as a plugin (actually a pair of plugins, see below) in the EAGLE portal. The EAGLE portal is a WordPress 4 installation and has an EAGLE-specific theme called "eaglenetwork" based on the Flexible theme by elegantthemes². The WordPress installation includes a number of open source plugins and a few other plugins that were written for EAGLE, most importantly the "Eagle Search Inscriptions" plugin. (For details about the EAGLE portal, the Eagle Search Inscriptions plugin and where to find the source code see D5.2.)

Several standalone alternatives for the FSA, including a TiddlyWiki³ and basing it on the storytelling application of the MOVIO system for virtual exhibitions⁴, were discussed and rejected. The decision to implement the FSA as a WordPress plugin reflects the goal of sharing the EAGLE corporate design as well as fully integrating the FSA with the other functionalities of the EAGLE portal. For example, the plugin can directly access the user accounts that are used throughout the EAGLE portal, and it can make use of the built-in WYSIWYG editor.

After the decision to implement the FSA as a WordPress plugin, it became clear that not much code could be reused form existing plugins. The most promising candidate was the "Ancient World Linked Data for WordPress" plugin⁵ that also collects data from different sources, displays them as popups on mouseover over supported URLs and can sort the supported URLs on a web page by type (e.g. place names) or source (e.g. Arachne). However, the focus is too different from the FSA and it does not do anything with the data. Still, it can be taken as an inspiration as to which data sources to implement.

Some technical background: WordPress organizes its content in objects called posts. A post has a list of attributes, such as title, author, publication status, and may contain additional optional metadata. Posts also have a type, such as page and blog post (which is also just called "post", as it was the initial intention of WordPress to be a blogging software), as well as other types that are of no interest here. It depends on the type of a post which template is used to display it in the frontend, which metadata it supports, and what functionality such as commenting or categories are enabled for this item. Displaying a page is similar to displaying a blog post, but pages are normally limited in number and tend to include fixed information such as an "about" page. In an unmodified WordPress installation the start page includes links to all pages and shows a paginated list of all blog posts.

² http://www.elegantthemes.com/gallery/flexible/

³ http://tiddlywiki.com/

⁴ http://www.movio.beniculturali.it/

⁵ https://wordpress.org/plugins/ancient-world-linked-data-for-wordpress/



The backend of a WordPress installation is a MySQL database. All posts are stored in a table called "wp_posts". In the data field "post_type" they have the values "post" and "page", respectively. Blog posts and pages are created on different but very similar internal "edit pages". The main difference is that the edit pages can include different sets of widgets for adding tags etc. The relevant URL parts are "/wp-admin/edit.php" for blog posts (the default post_type "post" is not spelled out) and "/wp-admin/edit.php? post type=page" for pages.

The EAGLE portal does not display blog posts on the start page ("Home"), but as items on the News page. Links to the News page as well as other pages are in the header area of the normal page template. The comment functionality is generally turned off in the EAGLE portal.

3.1.1. Two plugins

Verison 1.0 of the FSA was a single plugin. In order to make the FSA more easily reusable, its sourcecode is now split up into two separate WordPress plugins. The first plugin is called "Eagle Storytelling Application". It contains all generic functionalities of the FSA and works as a standalone plugin. The second plugin is called "Eagle Storytelling Application Bridge" and contains the code for a new "story" post type and the code that is specific to the EAGLE portal.

As usual the plugins are located in subfolders of the standard "wp-content/plugins" folder. In fact, all the files of the FSA plugins are in this folder. Even the template files for the FSA access page and the story pages are in the "Eagle Storytelling Application Bridge" plugin folder instead of the theme folder.

While the abbreviation FSA for the Flagship Storytelling Application was chosen in analogy to the acronym FMA of the Flagship Mobile Application, the namespace of the WordPress plugins is "esa", in analogy to the namespaces "et" of the EAGLE theme and "esi" of the EAGLE plugin for searching inscriptions. It is unlikely that this will lead to any confusion.

3.1.2. The Eagle Storytelling Application plugin

This plugin enables the search in the data sources via the "Add Media" modal window. The search results are displayed in a standardised way and can be inserted into pages and blog posts.

The WordPress search functionality is extended to include the content of the inserted items. The plugin adds two new tables to the WordPress database: "esa_item_cache" and "esa_item_to_post". The first table "esa_item_cache" contains the content of the items, so that it is not necessary to make an API call every time the item is displayed and to make the content of the items searchable in the story search. Primary key of this table is the combination of data source identifier (such as "wiki", "eagle", "europeana" etc.) and the ID of the item. The second table "esa_item_to_post" saves the information which stories contain which esa item shortcodes, so they can be found by searching in the esa item cache.

The "Eagle Storytelling Application" plugin works out of the box with any unaltered modern WordPress installation. It will be published in the WordPress plugin repository⁶ with the usual GPLv2 Licence for WordPress plugins and can then be installed via all usual methods. Right now, however, the plugin is still a beta version and can be installed only manually, i.e. the usual steps of downloading and unzipping the plugin, extracting it in the wordpress/plugins folder, and activating it in the admin area. In addition, one can specify the data sources one wants to use. The default is all available data sources.

3.1.3. The Eagle Storytelling Application Bridge plugin

This plugin creates the custom post type "story". Although the "story" post type is not EAGLE-specific, it is in this plugin and not the generic plugin because it strongly interferes with the standard settings of a

⁶ https://wordpress.org/plugins/



WordPress installation. The plugin limits the functionalities of the first plugin to this post type. It also adapts the user roles to the new "story" post type. When the plugin is disabled for maintenance, the "story" post type will disappear, but the changes to the user roles will persist. Otherwise the roles of the existing user accounts would be lost every time the plugin is disabled. (This is the only behaviour of the plugin that cannot be completely removed by deactivating it.) The plugin also introduces a separate fulltext story search and separate keywords for the "story" post type.

The EAGLE-specific part mainly consists of the adapted templates of the custom EAGLE theme. This includes the template for the sidebar with the story search and the EAGLE-specific search filters. The "keywords" and "authors" filters are restricted to stories, but are in principle generic. The "language" filter is new and normally not needed in a blog, but could also be regarded as fairly generic. Only the specific choice of filters makes them EAGLE-specific. Only the "Europeana" filter is truly EAGLE-specific.

The "Eagle Storytelling Application Bridge" plugin needs the EAGLE theme and the "Eagle Storytelling Application" plugin to function. It is only meant for the EAGLE portal and has to be installed manually. In addition, one has to create the empty page "stories" for the FSA access page. It would be possible to install the plugin also in other blogs, but then the EAGLE-specific code for the themes would need to be changed.

3.1.4. The "story" post type

The "Eagle Storytelling Application Bridge" plugin introduces a new blog post type called "story". The goal is to distinguish between EAGLE's News items and stories, and to separate the limited number of EAGLE project partners from the potentially unlimited list of story authors. In addition, new story authors get the WordPress user role of "subscriber", which does not have the right to write News items, but has been enhanced with the rights to write and publish stories.

Stories are created in yet another edit page /wp-admin/edit.php?post_type=story (for more details see section 3.4), and are also stored in "wp_posts". The story edit page is modified by the plugin to accommodate the needs of the FSA with a different set of widgets. In accordance with the policy of the EAGLE portal, it is not possible to comment on stories.

3.2. THE FSA ACCESS PAGE

The FSA access page is implemented as a page in the WordPress sense. It is comparable to the start page displaying a paginated list of blog posts, or in the case of the EAGLE portal the News page.

The URL path of the FSA access page is /stories/. It has its own associated template page "page-stories.php".

3.2.1. Design rationale

The FAS access page was designed to visually resemble the News page. The FSA access page shares the idea of searching and filtering the results with the Inscription Search page (with the obvious difference that the FSA access page searches stories and the Inscription Search page searches inscriptions), but since stories resemble news items more closely than they resemble inscriptions, the design of the News page is more appropriate for the FSA access page than the design of the Inscription Search page.

There are, however, some differences in the set of widgets in the sidebar between the News page and the FSA access page that arise from the specific needs of the FSA. For example, while the News page has a News archive sorted by months, this does not make much sense for user-generated stories. On the other hand, the FSA access page has an arsenal of filters for restricting search results that wouldn't make sense for the small number of news items.



3.2.2. The sidebar

The sidebar starts with a box "STORIES", which contains a link leading back to the FSA access page. The next two widgets, "SEARCH STORIES" and "FILTERS", are described in the next section.

The following widget is "LATEST STORIES". This is comparable to "LATEST NEWS" on the News page. It is based on the same search as for the list of all published stories, but only the titles of the five most recent stories are displayed.

The last widget is the "LOGIN" widget. The login itself is not specific to the FSA, and technical details about the login process are described in deliverable D5.2 about the EAGLE portal. Users that are logged in will also see a link "create new story" that leads to the edit view of the story editor, and another link "edit my stories" leads to the list view of the story where the user can see their published and unpublished stories.

3.2.3. Searching and filtering stories

The "SEARCH STORIES" widget searches the full text of all published EAGLE stories. The basic search functionality for blog posts, i.e. all entries in the table "wp_posts" with the normal post_type "post" and the post_status "publish", is provided by the WordPress engine. The "Eagle Storytelling Application" plugin enhances this search to include the text of the embedded items from external data sources. The "Eagle Storytelling Application Bridge" plugin in turn changes the search from blog posts to stories, i.e. it searches the entries with post_type "story" instead. If a search was performed, the list of all published stories in the main column is replaced by the search result.

The "Eagle Storytelling Application Bridge" plugin also introduces the search filters: The "FILTERS" widget is actually a group of filter widgets called "Filter by keyword", "Filter by author", "Filter by Europeana ID" and "Language". All of these filters take the search result (or all published stories if the user did not perform a search) and filter this list for additional criteria. That is to say, only the stories that also fulfil the additional criteria will be displayed.

In the first of these filters, "Filter by keyword", the user can select a keyword from a word cloud. It is not possible to search for a keyword if it not included in the word cloud. (However, one can change the "term" parameter in the URL directly.) The design decision against a freetext search for keywords and to allow only one keyword at a time is discussed in section 4.1.3.

The second filter is "Filter by author", where author means user name. It adds the criteria that the data field "post_author" in the table "wp_posts" should match the name that has been entered by the user. The filter is implemented as a dropdown list.

The third filter "Filter by Europeana ID", where the user can filter the search result for stories that contain inscriptions with a given Europeana ID, is implemented differently as there is no corresponding data field that lists the Europeana IDs of these inscriptions. The filter searches for stories that contain a Europeana shortcode (see section 3.4.2) with the given ID.

The last filter "Language" is implemented as a dropdown menu. It is more resilient than the other filters. For example, if a language is selected and the user clicks on a keyword of a story, the fulltext search and the other filters are set to default, but the selected language remains selected.

3.3. THE STORY PAGES

The story pages are comparable to pages displaying a single blog post. When the user publishes a story, it gets a unique URL (a "pretty URL", see below) and the corresponding story page is created automatically by the WordPress engine, or more precisely, the page will be created on-the-fly when the URL is requested. The URL path of a single story page is /stories/ plus a sanitized version of title of the



story that is stored in the data field "post_name" in "wp_posts". For example, the story with the title "Funerary Inscriptions in the current EAGLE collection" has the URL path

/story/funerary-inscriptions-in-the-current-eagle-collection/

It is not possible to use the same /stories/ as the FSA access page since this would confuse WordPress, hence /story/.

Since stories are blog posts with a different post_type, there is no explicit page counterpart of the FSA access page. However, the "Eagle Storytelling Application Bridge" plugin makes sure that the story pages use the dedicated template file "page-stories-single.php" by changing the default template file for the post_type "story". The FSA access page is not affected by this.

The design of the main column is similar to the template for news items, again with the difference that only posts with the post_type "story" are displayed. Another difference is that the excerpt of a story, if there is one, will be displayed in italics below the title and before the main text of the story.

The sidebar is identical to the sidebar in the FSA access page. Searching and filtering always leads back to a result list on the FSA access page since they only make sense on a list of stories rather than a single story. The only difference to the FSA access page is that logging in and out of course leads back to the story page.

3.4. THE STORY EDITOR

The generic "Eagle Storytelling Application" plugin adds the search of external data sources and the shortcode mechanism to the standard blog post editor. The "Eagle Storytelling Application Bridge" plugin creates the separate story editor.

3.4.1. Searching EAGLE inscriptions and other data sources

The search in external data sources provides a consistent user interface for the simple searches in these data sources. It does not make sense to replicate the complete search capabilities of the external data sources. After all, the user can always search in the original data source and copy the URL into the FSA search. The FSA will then turn the URL into a shortcode by analysing the URL of the item.

The implementation of the data sources is modular. There is an interface (i.e. an abstract class) where one can add new data sources with little effort in a one separate file per data source. Data sources can be implemented if they have a search API and unique URLs or at least unique IDs for each item. For example, in EAGLE itself the FSA has access to the solr index of the EAGLE portal (see D5.2), i.e. the search API is the standard solr API, and there are no unique URLs but unique IDs for the EAGLE items.

Some notes about the implementation of the individual data sources:

Europeana: There are additional filters for the media type in Europeana, and a checkbox to restrict the search to EAGLE items in Europeana. One difference to searching EAGLE directly is that EAGLE items do have unique Europeana URLs.

iDAI.gazetteer: There are no additional filters, just as in the simple search of the original data source. The iDAI.gazetteer distinguishes between the URL representing a *place* and the URL of the electronic *representation* of the place in the system. The FSA accepts both types of URLs.

Wikipedia: There are no additional filters. In particular, there is no filter for the language yet, and a search is carried out in the English Wikipedia. However, the system accepts Wikipedia URLs in all languages. Only the introductory paragraph of the Wikipedia entry is displayed in the FSA.

EAGLE: As already mentioned, the EAGLE search in the FSA uses the solr index directly. While the solr index provides a search API, handling the result was more difficult to implement than other data sources



due to the very complex EAGLE XML format.

The EAGLE search within the FSA seems to be more important than implementing the requirement STE16, i.e. letting the user save an item in their private space within the EAGLE inscription search and then letting them copy it into the FSA. On the other hand, the FSA does not faithfully reproduce even the simple search interface in the EAGLE inscription search since the interface design is very different from the FSA and from the rest of the EAGLE portal.

YouTube: WordPress already supports YouTube directly. YouTube URL can be pasted into the the editor and will automatically be displayed in the usual way. There was no point in replicating this with a shortcode within the FSA. As a consequence, the method for embedding YouTube videos is a little different from the other data sources. Also, the text from the YouTube page of the video is not searchable in the story search, and the video still image cannot be selected as the thumbnail image of the story. However, there seems to be no need for that.

3.4.2. Shortcodes

WordPress has the concept of so-called shortcodes such as "[inscription id="12345"]". This abstraction makes it possible to embed items from external sources in the same way as simple links. The items are cached in the database every time the story is saved. The caching of the texts that belong to the embedded items enables the story search to include these texts too.

Examples for shortcodes in the FSA are:

EAGLE: [esa source="eagle" id="DAI::dae53bf85e7f6566cce12106309fa4c6::artifact" height="100"]

Europeana: [esa source="europeana" id="(very long ID)"]

iDAI.gazetteer: [esa source="idai" id="2282601"]

Wikipedia: [esa source="wiki" id="Wheel"], with the possibility to specify a language such as Japanese: [esa source="wiki" id="車輪@ja"]. The default language is English.

Shortcodes can also be inserted directly into the story text. The system will immediately loads the item content from the original source. The system caches the item content as the next time the story is saved. After that, every time the story is displayed for reading or for editing, the system checks whether the cached version is older than tow weeks. In this case it loads the item anew from the original data source.

3.4.3. The separate story editor

The "Eagle Storytelling Application Bridge" creates the separate story editor with a list view and an edit view as a clone of the normal WordPress edit page, but dedicated to stories. The list view has the URL path

/wp-admin/edit.php?post_type=story

The edit view has the URL path

/wp-admin/post-new.php?post_type=story (new story)

/wp-admin/post.php?post=...&action=edit (existing story or draft)

where "..." denotes the story ID. Thus, stories are referred to by their IDs rather than their "pretty URL" as on the story pages. This makes sense especially for new stories that do not yet have a title to generate a pretty URL from. On the other hand, the URL path "/wp-admin/post.php?post=..." without the "&action=edit" will be automatically replaced by the pretty URL of the corresponding story page.

Only the story author and the administrators can edit a story. As with a normal blog post, the user can enter keywords and an excerpt for the story, save drafts and publish the story. A story draft normally has



the post_status "inherit", a published story has the post_status "publish". All of this is provided by the WordPress engine "for free" as soon as the clone of the normal edit page has been created. Thus, the requirements STE05, STE06 and STE07 that deal with the behaviour of the story editor are automatically provided by WordPress (see section 2.3.2 and 2.3.3 for a discussion of these requirements).

The sidebar of the edit view contains a "Keywords" widget, which is the counterpart of the "Filter by keyword" widget on the FSA access page. The implementation is again provided by the WordPress engine. In addition, the FSA plugin separates the story keywords from the keywords for news items.

There is no counterpart for the "Filter by author" widget as the author cannot be changed by the user. The user who creates the story is automatically the author. The "Filter by Europeana ID" widget has no counterpart in the edit view either.



4. CONCLUSIONS AND FUTURE PLANS

The Flagship Storytelling Application (FSA) has been officially launched on September 24, 2015 on occasion of the Sixth EAGLE International Event "Off the Beaten Track. Epigraphy at the Borders" in Bari. Even before its official launch, a lot of remarkable stories had already been created with the FSA, but with the multiplicators in Bari spreading the message to the EAGLE partners and the interested public, we expect a whole wave of fascinating new stories. The FSA is prepared for it.

The version 2.0 of the FSA that was presented in Bari has come a long way since the version 1.0 described in Deliverable D5.4.1. In general, we have created a more powerful and intuitive story editing environment for authors. One major addition was the search of external data sources: Multimedia content such as inscriptions, maps, objects and monuments can now easily be inserted into the stories. We also added featured images for stories (thumbnails), which was a feature request after the initial presentation of the FSA. Likewise, the search filter functionality for stories was significantly enhanced with filters for authors, keywords and Europeana IDs.

We do not envisage a specific target audience for the FSA. Instead, we see the FSA as a tool for inscription-related stories on all levels. Some stories may be interesting for the general public, while others may appeal only the specialists. We welcome everyone as reader, and we don't think that we should make specifications about the complexity of the stories. However, it may make sense to mark stories with terms such as "easy", "intermediate" and "specialist", or the FSA could provide different starting points for readers with different interests rather than always starting with a list of the most recent stories. Another possibility would be "stories for tourists", i.e. a map of stories according to the gazetteer entries they contain, or a list of all stories that include inscriptions from their current location. This could be implemented as a search for stories that contain a gazetteer entry of the present location, or a search of the location field of whole EAGLE database. We will explore these possibilities in the following months.

4.1. DESIGN GOALS IN VERSION 2.0 AND BEYOND

After version 1.0 we have focused on a few important design goals: consistency, encapsulation, ease of use and the findability of stories. Great progress has been made on these goals in version 2.0, and we continue to work on this. In addition, a main focus in the last 6 months of the EAGLE project will be the sustainability of the FSA.

4.1.1. Consistency

Consistency refers to the integration with the EAGLE portal as well as a consistent user interface within the FSA. The biggest step towards the integration with the EAGLE was the decision, made even before version 1.0, to implement the FSA as a WordPress plugin. One immediate result was that the same user credentials could be used in the EAGLE portal and the FSA. In the design of the FSA access page and the individual story pages we have also taken care to be consistent with the design of the EAGLE portal.

When the private user space is used as a connection between the EAGLE inscription search and the FSA (for a detailed discussion see section 4.2), there will be a synergy between these two services. However, using the private space as a replacement of the list view in the story editor would only work if the design of the private space as part of the EAGLE inscription search would be aligned with the design of the FSA, and we don't see an easy way of aligning it with the FSA design or even the design of the external data source search. In any case the design would certainly need to be adapted on both sides. In line with the other external data sources, we also only implement the simple EAGLE inscription search. One low-priority goal is to standardise at least the login process, which currently a specific login page in the FSA and a modal window in the EAGLE inscription search.



Within the FSA, we have attempted to make similar things look similar. For example, the design of the search in different data sources and the presentation of the results is as similar as possible. Internally, the shortcodes for the the data sources are again a means of standardising the process of including items from these sources. And even though the text associated to such an item is not part of the story, the story search treats it as if it were.

Another example is the way the author can choose a thumbnail from the images in the story text: While it looks the same to the author, internally this is a combination of the standard WordPress "featured image" mechanism and the handling of the items from the external data sources.

4.1.2. Encapsulation

Encapsulation is the counterpart of the consistency with the EAGLE portal. It means that the FSA and the EAGLE should not interfere where it is not wanted, and also that the code of the FSA is encapsulated from the rest of the EAGLE portal.

The main goal was the possibility for new users to create stories, but not News items. This was a necessary step before the FSA could be opened to a wider circle of users beyond the EAGLE project partners. The current solution is simply to give the "lowest" of the built-in WordPress, users roles (subscriber) the right to write and publish stories. This seems to be simpler than creating an explicit user role such as "story author". However, if it turns out that some story authors should not have the right to publish their stories without a review of the story, there may be the need to create user roles such as "story author" and "story contributor".

Another point where the EAGLE portal and the FSA interfere is the modal window for the search of external datas sources. It is not quite clear whether e.g. News items are supposed to be able to access this search, and vice versa it is unclear whether the FSA should share the media library with the EAGLE portal. On the other hand, this problem only occurs when using both plugins (see section 3.1.1). If only the generic plugin is used, then it is the desired behaviour that there is no separate modal window for the search of external data sources.

The encapsulation of the generic parts and the EAGLE-specific parts of the FSA into two different plugins will be discussed in section 4.1.5 below.

Stories are at the moment stored in the table "wp_posts", together with all news items, pages and projects. At the moment, entries in "wp_posts" are created by a small number of EAGLE affiliates.

4.1.3. Ease of use

We have worked on a more powerful and intuitive story editing environment for authors, as well as making the FSA self-explaining to story readers. Ideally, both authors and readers should be able to use the FSA without ever having to consult this Deliverable, and authors should need no knowledge of WordPress to write their stories.

For story readers, we made it easier to navigate through the stories (see the next section). We have also simplified the search filters, for example by turning the author filter into a list of existing authors rather than a freetext field, and by leaving out the keyword freetext field and allowing only one keyword at a time. (We might implement freetext filters with autocomplete of the search term, similar to Wikipedia.) Technically it would be easy to implement filtering for more than one keyword and to implement even more filters, but it is clear that no one is going to spend much time to learn the details of the story search. If the search becomes too complex, people will simply use Google instead. After all, the FSA is not comparable to tools such as the iDAI.objects database (aka Arachne) for archaeologists where it makes sense to spend some time learning how to use its search facilities and in return getting better search results than via Google. The point was to make the story filters obviously useful. However, filters can still



be used together. For example, one can set a language and still search for a keyword.

If the items from an external dat source are very long, they have a long and a short view. The short view is the default, and the reader can toggle between these views.

We have also added featured images for stories (i.e. thumbnails) in the list of stories, which was a feature request after the initial presentation of the FSA. This makes it easier to decide whether one wants to read a story.

For story authors, we have rearranged the editor parts to resemble the design of the finished story more closely. For example, we have moved the excerpt field on top of the main text field. The "Keywords" widget may also be moved to a position below the title field.

In a second step, it would be nice to redesign the story editor page to resemble the design of the FSA access page rather than a modified WordPress blog post editing page. This could be combined with a new URL scheme that is more independent of the Wordpress URL scheme and would replace especially the URLs of the type "/wp-admin/post.php?post=...&action=edit" when editing stories. And perhaps the list view of all stories by one author could then be replaced by a simple search result page for stories written by the user. There could be a new button "also show my drafts" and a possibility to show just the titles without excerpts on the FSA access page.

With the introduction of shortcodes, we are also catering for different proficiency levels of the story authors. If an author wants to insert items from the supported external datasets but prefers not to work with the search modal window, they can directly insert a shortcode into the story text (in the "text" mode of the editor). On the other hand, an even easier method would be to allow to paste the URLs of the item directly into the text, in the same way WordPress supports YouTube links out of the box. We are currently evaluating if this can be done without too much effort within a plugin. And this method only works with data sources have a REST interface where each item has indeed a unique URL. Especially EAGLE itself does not have this.

4.1.4. Findability of stories

The Findability of stories is already part of the requirements, which demanded fulltext search, a keyword cloud and a list of the latest stories. However, it soon became clear that this is not enough to navigate through a large number of stories. We have added obvious filters for the author name and the language of the story. Authors have the same problem as keywords that one cannot easily guess an author's name if one doesn't know in advance that a specific person has written a story. On the other hand, an author wordcloud didn't seem appropriate, and therefore the filter is implemented as a dropdown list of the authors with the most stories. Likewise, the language filter is a dropdown menu. A Europeana filter allows to search for objects with a given Europeana ID, although we expect this filter to be rarely used because the format of the Europeana IDs is very specific, and one cannot guess valid Europeana IDs. We will add (or replace the Europeana filter by) a filter for Trismegistos IDs as soon as there is a critical mass of EAGLE inscriptions with a Trismegistos ID.

In the main column, the author names and keywords related to a story are now klickable links. In the main text, all items that have been inserted via the search of external data sources have a button "Find stories with this item". And text from the items is included in the fulltext search even though the story text itself contains only the shortcode such as [esa source="wiki" id="Wheel"] rather than the whole text.

We are also planning to include links for terms in the EAGLE vocabulary. A word auch as "marble" could trigger a search for all stories that contain this term, and a popup on mouseover could reveal the available translations and a link to the vocabulary page for "marble". In the same way the keywords could be multilingual, so that "marble" would also find "Marmor" in stories written in German. However, it is not yet clear whether for example all occurrences of "marble" should be clickable links because "marble" is in



the EAGLE vocabulary, or whether words need to be explicitly tagged, i.e. in the same way as internal Wikipedia links where a word or phrase is normally only tagged the first time it occurs in the text. The automated approach could lead to too many links in the text and to false positives (the tagged word means something else) and false negatives (the word is not tagged because it is spelled differently, for example "marbles"). On the other hand, tagging words by hand would take a long time.

Another idea for navigating through the stories would be to introduce shortcuts for the stories themselves, which could then be inserted into other stories. Whether this would be a better or easier solution than normal links to other stories remains to be seen.

4.1.5. Sustainability

One main focus of our work in the remaining six months of the EAGLE project will be on making the FSA more sustainable. On the one hand, we will make sure that the FSA will be administrable with minimum effort as part of the permanent hosting of the EAGLE portal after the end of the project. For example, we have introduced a clean sourcecode design with an interface (i.e. an abstract class) where one can add new data sources with little effort.

On the other hand, we want to make sure that the FSA can be used independently of the EAGLE portal. To this end, we have split the FSA sourcecode into two plugins, namely a generic part and an EAGLE-specific part. The generic plugin will be uploaded to the standard WordPress plugin repository. The EAGLE-specific part consists mainly of the custom changes to the "eaglenetwork" theme of the EAGLE portal (see section 3.1). This split has several advantages: It reduces the maintenance effort even within the EAGLE-specific part, and, more importantly, it makes the generic part of the FSA reusable in other WordPress blogs. The split is fairly recent, and some EAGLE partners have agreed to test the generic plugin in their own blogs. If there is demand from the beta testers, we may even put everything related to the "story" post type, now part of the EAGLE-specific plugin, into a third plugin.

All the efforts for sustainability go beyond what was originally required from the FSA.

4.2. **VERSION 3.0**

The FSA in its current version 2.0 and without any further development can already be called a success. However, the FSA has taken on a life on its own, and in the remaining project time we are planning to make the FSA even more useful. The final version 3.0 of the FSA will be available by the end of the EAGLE project.

Some possible features of the final version 3.0 have already been discussed in the previous section. Here we describe the remaining original requirements. With very few exceptions, all requirements for the FSA within the EAGLE project have already been implemented. As with every software project, the goals, requirements and priorities have changed since the initial requirements were formulated. We intend to implement the missing requirements in version 3.0 if implementing them still makes sense and is at all possible. However, since the FSA is already perfectly usable and our focus in the remaining time is on sustainability, we might skip some of these requirements if there is not enough real demand from the EAGLE communities for these features.

Perseus as a data source (STE09) states that users should be able to search and include passages of Ancient Greek and Roman texts from the Perseus Digital Library. This will be implemented as soon as the Perseus API is stable enough (we are in contact with Perseus about that). We hope that this will be the case in time for version 3.0.

Accessing saved items (STE16) states that users should be able to access items stored in their personal space and include them into their stories. In their private space, a user can save search results and individual items from the EAGLE collection (for more details see D5.2). STE16 has turned out in practice



to be a low-priority goal, but it should be straightforward to implement. However, it needs more discussion with the EAGLE inscription search team about the right way to integrate this with the FSA. In particular, the integration is a two-way road and it should be possible to save stories (i.e. posts with "story" post type) in the private space as well, and then references to saved stories might also be embeddable in other stories.

A corresponding requirement for the EAGLE portal is PSE28, which is mainly about the private space in general, but states in a note that users should be able to view the list of the stories created by them in their personal space (see D5.1). In other words, the user will be able to access the list view of the story editor in their personal space. If this side requirement will be implemented, the list view of the story editor might disappear completely if it is no longer needed.



5. REFERENCES

Note: All EAGLE Deliverables are listed at http://www.eagle-network.eu/about/documents-deliverables/

[D5.1]

F. Mambrini. EAGLE portal and services requirements and specifications. EAGLE Deliverable 5.1. Available online at:

http://www.eagle-network.eu/wp-content/uploads/2013/06/EAGLE D5.1 -Portal-and-services-requirements-and-specifications.pdf

[D5.2]

C. Prandoni, N. Alfarano, V. Casarosa. EAGLE Portal. EAGLE Deliverable 5.2. Available online at: http://www.eagle-network.eu/wp-content/uploads/2013/06/EAGLE D5.2 EAGLE-Portal v1.0.pdf

[D5.4]

X. Martin. First Release of the Flagship Mobile Application and SDK. EAGLE Deliverable 5.4. Available online at:

http://www.eagle-network.eu/wp-content/uploads/2013/06/EAGLE_D5.3.1_First-release-of-the-flagship-mobile-application-and-SDK_v1.0.pdf

[D5.5]

F. Mambrini. Validation Plan. EAGLE Deliverable 5.5. Available online at:

http://www.eagle-network.eu/wp-content/uploads/2013/06/EAGLE D5.5 Validation-Plan v1.0.pdf

[D5.6]

F. Mambrini. Validation report. EAGLE Deliverable 5.6. Available online at:

http://www.eagle-network.eu/wp-content/uploads/2013/06/ EAGLE D5.6.1 Validation Report First Release v1.0.pdf

[D6.1]

C. Prandoni, R. Santucci, N. Clonini. EAGLE Project Website. EAGLE Deliverable 6.1. Available online at:

http://www.eagle-network.eu/wp-content/uploads/2013/06/EAGLE_D6.1_EAGLE-Project-Website v1.0.pdf